

# Discretization of a Model for the Formation of Longshore Sand Ridges

Juan Mario Restrepo

Mathematics and Computer Science Division

Argonne National Laboratory, Argonne IL U.S.A. 60439

Jerry L. Bona

Mathematics Department

The Pennsylvania State University, University Park PA U.S.A. 16802

**Subject Classification:** 86-08, 86A05.

**keywords:** sand ridges, predictor-corrector, fixed point, finite differences.

Preprint MCS-P408-1293,

Mathematics and Computer Science Division

Argonne National Laboratory, Argonne, IL, 1993

### **Abstract**

This paper presents and evaluates the numerical solution of a coupled system of equations that arises in a model for the formation and evolution of three-dimensional longshore sand ridges. The model is based on the interaction between surficial or internal weakly nonlinear shallow-water waves, having weak spanwise spatial dependence, and the deformable bottom topography.

The presentation of the details concerning the discretization of the model is primarily motivated by two facts: (1) The model involves equations for which little is known regarding its solutions, and (2) the predictor-corrector scheme presented here, which combines finite difference techniques and fixed-point methods, is simple, fast, and general enough to be used in the discretization of partial differential equations with local nonlinearities whose solutions are smooth.

# 1 Introduction

The following system of equations constitute a model for the formation and evolution of three-dimensional sand ridges on the continental shelf:

$$\begin{aligned}
 a_{1x} - iK_1 a_{1yy} + iK_3 f(x, y) a_1 + iK_5 e^{-i\delta x} a_1^* a_2 &= 0 \\
 a_{2x} - iK_2 a_{2yy} + iK_4 f(x, y) a_2 + iK_6 e^{+i\delta x} a_1^2 &= 0 \\
 a_1(x = 0, y) &= \mathcal{A}_1(y) \\
 a_2(x = 0, y) &= \mathcal{A}_2(y),
 \end{aligned} \tag{1}$$

and

$$\begin{aligned}
 \frac{\partial}{\partial T} h(x, y, T) &= \frac{K}{\rho_0} [\mu_x(a_1, a_2) + \nu_y(a_1, a_2)] \\
 h(x, y, 0) &= \mathcal{H}(x, y),
 \end{aligned} \tag{2}$$

with appropriate boundary conditions on  $y = 0$  and  $y = N$ . The real constant coefficients  $K$  are  $O(\alpha, \varepsilon)$ , where  $\alpha$  characterizes the degree of nonlinearity of the waves and  $\varepsilon$  the size of the slopes of the bottom topography. The constants are given in the appendix. Equation (1) describes the spatial structure of the complex amplitudes of the two most energetic modes of weakly nonlinear dispersive ocean waves traveling in the shoreward direction  $x$  over a deformable bottom topography  $h = 1 + \varepsilon f(x, y)$ . The variable  $y$  is the spanwise coordinate. The bottom evolution obeys the mass transport relation, Eq. (2). A full description of the model appears in [1] and [2].

Owing to the widely discrepant time scales between the (fast) evolution of the water waves and the (slow) bottom topography, this coupled system may be solved iteratively: Given an initial bottom configuration  $\mathcal{H}(x, y)$ , we seek a solution to the

water waves using Eq. (1). The bottom is then updated using the mass transport equation, and the whole procedure is repeated with the new bottom until some prescribed time.

The similarity of Eq. (1) to the nonlinear Schrodinger equation (NLSE) leads us to conjecture that the numerical technique presented here applies to the NLSE in a straightforward manner. In fact, any equation or system with solutions of sufficient regularity with local nonlinearity may be solved by our method.

Several issues have motivated the particular choice of the scheme to be presented: (1) an efficient, simple, and sufficiently accurate method is desired to implement the above nonstiff, nonlinear system numerically; (2) the accuracy requirements are not very sophisticated since the main objective is the exploration of phenomenological questions; (3) a uniform grid is preferred over a variable one, so that both the surface and mass transport equations may be easily computed on the same grid; and (4) the computational domain is fairly large for the sort of problem presented in this study. We refer to the numerical scheme adopted in this study as the fixed-point method (FPM). Among its best features are low storage requirements and high speed.

The input to the model is composed of an initial bottom configuration and the mode amplitudes at the line  $x = 0$ . The required dynamical parameters are the fundamental frequency, which is needed to determine the  $K$  constants; an estimate of the size of  $\alpha \ll 1$  and  $\beta \ll 1$ , respectively giving the degree of nonlinearity and dispersiveness in the waves; and the dimensions of the rectangular patch,  $0 \leq x \leq M$ ,  $0 \leq y \leq N$ , of ocean on which the solution is to be computed.

The following difference operators pertain to the discretization:

$$\begin{aligned}
\Delta_q &= u(q_{j+1}) - u(q_j) && \text{forward difference} \\
\nabla_q &= u(q_j) - u(q_{j-1}) && \text{backward difference} \\
\delta_q &= u(q_{j+1/2}) - u(q_{j-1/2}) && \text{central difference} \\
A_q &= u(q_{j+1}) + u(q_j) && \text{forward average}
\end{aligned} \tag{3}$$

in the independent variable  $q$ , say. The physical space is given by  $\mathcal{R}^2 \times T \equiv [0 \leq x \leq M, 0 \leq y \leq N] \times \{T \geq 0\}$ . Define  $\mathcal{R}_\Delta^2 \times T_\Delta \equiv (x_r, y_s) \times T_n = (r\Delta x, s\Delta y) \times n\Delta T \in \mathcal{R}^2 \times T$ . Furthermore, there are integers  $m$  and  $n$ , such that  $M = m\Delta x$ ,  $N = n\Delta y$ .

## 2 Discretization of the Mass Transport Equation

The mass transport equation is implemented numerically by using a two-step Lax-Wendroff scheme, which is second-order accurate in time and space. Since this technique is very well established [3], we shall not report on such standard issues as consistency, convergence, and uniqueness.

Equation (2) is approximated by the following computational module:

$$\begin{aligned}
h_{r+1/2}^{n+1/2} &= \frac{1}{4}(A_x + A_y)h + \frac{\Delta T}{2\Delta x}\Delta_x\mu + \frac{\Delta T}{2\Delta y}\Delta_y\nu \\
h_j^{n+1} &= h_j^n + \frac{\Delta T}{\Delta x}\delta_{xT}\mu + \frac{\Delta T}{\Delta y}\delta_{yT}\nu
\end{aligned} \tag{4}$$

on  $\mathcal{R}_\Delta^2 \times T_\Delta$ .

For the sake of clarity, the stability criterion is established in the shoreward direction only. Since  $\mu_x = \mu_h h_x$ , substituting in Eq. (4) yields

$$h_j^{n+1} = h_j^n - \xi_x \left[ \frac{1}{2} (\Delta_x + \nabla_x) h - \frac{1}{2} \xi_x (\Delta_x - \nabla_x) h \right], \quad (5)$$

where  $\xi_x = -\mu_h \Delta T / 2\Delta x$ . A local stability criterion may be established by replacing  $h = \zeta^n \exp(ir\Delta x)$  in Eq. (5), from which it follows that the growth factor is such that

$$|\zeta|^2 = 1 - \xi_x (1 - \xi_x^2) (1 - \cos(r\Delta x))^2, \quad (6)$$

and formal linearized stability shall result if  $|\zeta| \leq 1$ , which restricts  $\xi_x^2 \leq 1$ .

Using the same argument, we can find the stability criterion in the spanwise direction, so that the stability of Eq. (4) in two space dimensions requires that  $\xi = -(\mu_h h_x \Delta T / 2\Delta x, \nu_h h_y \Delta T / 2\Delta y)$  be less than one in component form. Since

$$\begin{aligned} \mu_h &= -\sum_{j=1}^2 \frac{4\beta^2 k_j^3}{\mu_j \omega_j} (1 - \beta^2 h^2 k_j^2 / 2) H_j |a_j|^2 \\ \nu_h &= -\sum_{j=1}^2 \frac{4\beta^2 k_j h}{\mu_j \omega_j} (1 - \beta^2 h^2 k_j^2 / 2) J_j [I_j R_{jy} - I_{jy} R_j], \end{aligned} \quad (7)$$

where  $I_j$  and  $R_j$  are, respectively, the real and imaginary parts of  $a_j$ , it is possible to show that the maximum value attained by either  $|\mu_h|^2$ , or  $|\nu_h|^2$ , is of the order of  $16\beta^3 |a_j|^4$ . Hence, for stability the grid size is determined by the constraint

$$\frac{\Delta T}{\Delta x} \leq \frac{1}{4} \beta^{3/2} |a_j|^2, \quad (8)$$

a result that sits well with the need to be economical about computer resources and that does not conflict with the stability criterion of the overall iterative scheme of the full model. Thus, in component form, for  $\xi_x \leq 1$ , and assuming  $|a_j| \leq 1$  over the whole domain, we obtain

$$\frac{\Delta T}{\Delta x} \leq \beta^{-3} \alpha, \quad (9)$$

and for  $\xi_y \leq 1$ , the same argument leads to

$$\frac{\Delta T}{\Delta y} \leq \beta^{-3}. \quad (10)$$

Dissipation is known to occur except when  $\xi_x^2 = 1$ . The effect, however, can be quite small—fourth order in  $\Delta x$ —if the wavelengths are restricted to being much greater than the grid size.

## 3 Solution of the Surface Equations

### 3.1 Numerical Solution of the Two-Dimensional System

The two-dimensional case is considered here because its numerical solution will be used subsequently to evaluate the performance of the three-dimensional case. The two-dimensional case has been considered in detail by Bona et al. [4]. We adopt the same computational scheme and rely on their confidence in the method to enable us to use its results for comparison in the three-dimensional case. The

numerical solution of the two-dimensional wave system

$$\begin{aligned}
a_{1x} &= -iK_3f(x)a_1 - iK_5e^{-i\delta x}a_1^*a_2 \\
a_{2x} &= -iK_4f(x)a_2 - iK_6e^{+i\delta x}a_1^2 \\
a_1(x=0) &= \mathcal{A}_1 \\
a_2(x=0) &= \mathcal{A}_2,
\end{aligned} \tag{11}$$

where  $\mathcal{A}_1$  and  $\mathcal{A}_2$  are constants, will be used later in the evaluation of the implementation of the three-dimensional surface system. To solve the system, we use a standard fourth-order explicit Runge-Kutta scheme

$$\begin{aligned}
\phi_{r+1} &= \phi_r + \frac{1}{6}(\mathbf{P}_1 + 2\mathbf{P}_2 + 2\mathbf{P}_3 + \mathbf{P}_4) \\
\mathbf{P}_1 &= \Delta x \mathbf{F}(\phi_r, x_r) \\
\mathbf{P}_2 &= \Delta x \mathbf{F}(\phi_r + \frac{1}{2}\mathbf{P}_1, x_r + \frac{1}{2}\Delta x) \\
\mathbf{P}_3 &= \Delta x \mathbf{F}(\phi_r + \frac{1}{2}\mathbf{P}_2, x_r + \frac{1}{2}\Delta x) \\
\mathbf{P}_4 &= \Delta x \mathbf{F}(\phi_r + \mathbf{P}_3, x_r + \Delta x),
\end{aligned} \tag{12}$$

where  $\mathbf{F}$  is the right-hand side of Eq. (11), and the vector  $\phi_r \equiv [a_1(x_r), a_2(x_r)]$ . The details on the applicability of such a scheme to the solution of Eq. (11) appear in [4].

### 3.2 Numerical Solution of the Three-Dimensional Case

For the surface equations in the three-dimensional case, Eq. (1) is rewritten as

$$\begin{aligned}
a_{1x} - iK_1 a_{1yy} + iK_3 f(x, y) a_1 &= -iK_5 e^{-i\delta x} a_1^* a_2 \\
a_{2x} - iK_2 a_{2yy} + iK_4 f(x, y) a_2 &= -iK_6 e^{+i\delta x} a_1^2 \\
a_1(x = 0, y) &= \mathcal{A}_1(y) \\
a_2(x = 0, y) &= \mathcal{A}_2(y) \\
a_{1y}(x, y = 0) &= 0 \\
a_{1y}(x, y = M) &= 0 \\
a_{2y}(x, y = 0) &= 0 \\
a_{2y}(x, y = M) &= 0
\end{aligned} \tag{13}$$

to separate the linear and nonlinear parts. The first two boundary conditions are inherent in the physics of the problem. The remaining boundary conditions are artificial. These Neumann boundary conditions, combined with a computational procedure that will be explained presently, ensures that the overall structure of the solutions remains negligibly affected by the choice of lateral boundary conditions. We call this technique the “zero-flux procedure.”

To justify the need for such procedure, we spell out what sort of problem we are faced with: Since we need to compute a solution over a finite but large domain, care must be exercised in imposing boundary conditions on the lateral sides so as to avoid the introduction of structure in the solution that is strictly mathematical rather than physical in nature. A possible way to compute a solution of the problem over an effectively unbounded domain over a finite grid is to impose periodic

boundary conditions. However, periodicity imposes unwanted symmetries on the structure of the computed solutions. To avoid this situation, we use appropriate boundary conditions along the lateral sides and, in addition, place restrictions on the initial bottom configuration and the boundary condition at  $x = 0$  so that we can compute an oceanic event on a swath of what amounts to be an effectively unbounded domain. We have found that this zero flux procedure is superior to other synthetic boundary conditions in minimizing unwanted structure in the solutions.

The Neumann boundary conditions make the problem well-posed; however, by themselves, they would introduce a great deal of structure. Physically these boundary conditions correspond to placing hard barriers on the lateral sides of the domain. A posteriori we know that the solution to the model is two dimensional if neither the bottom nor the boundary condition at  $x = 0$  has  $y$  dependence. In such a case the zero flux condition on the lateral sides has no effect on the solution over any part of the domain (i.e., it does not lead to  $y$ -dependent solutions). We calculate the system over a computational domain that we divide into three regions. The large central region, flanked by two sufficiently wide lateral strips, is one in which  $y$  variation in the initial bottom or in the boundary condition at  $x = 0$  is possible. In the lateral strips no  $y$  dependence in the above-mentioned quantities is permitted. The solution in these lateral strips is discarded. The initial bottom and the boundary condition at  $x = 0$  are connected smoothly in all three regions so that a minimal amount of structure is introduced in the solutions. The size of the lateral strips is determined by what amounts to an educated guess.

Define the following vectors, with the superscript  $T$  meaning transpose, with

all  $K$ 's real:

$$\begin{aligned}
\mathbf{k} &= i[K_1, K_2]^T \\
\mathbf{k}_f &= if(x, y)[K_3, K_4]^T \\
\phi &= [a_1(x, y), a_2(x, y)]^T \in \mathcal{C}^2
\end{aligned} \tag{14}$$

with  $(x, y) \in \mathcal{R}_\Delta^2$ , so that the system, Eq. (13), is now recast on the discrete grid  $\mathcal{R}_\Delta^2$  as

$$[\partial_x - \mathbf{k}\partial_{yy} + \mathbf{k}_f]\phi = \mathbf{b}(x, y, \phi), \tag{15}$$

with the linear part on the left-hand side and the nonlinear terms on the right of the equals sign, plus boundary conditions,

$$\begin{aligned}
\phi_y &= 0 \quad \text{on } y=0, y=N, \\
\phi &= \phi_0 \quad \text{on } x=0, .
\end{aligned} \tag{16}$$

The term  $\mathbf{b}(x, \phi)$  represents the nonlinear terms. Succinctly, the above equation may be written as

$$\mathcal{L}\phi = \mathbf{b}, \tag{17}$$

where  $\mathcal{L}$  is the linear operator. Let  $L$  be a suitable discretization of the linear operator. Suppose the value of the vector  $\phi$  at level  $r$  for all  $s$  is known. Making use of fixed-point methods, we can find the value of the vector at level  $r + 1$ . Computationally, the calculation is performed in two steps: let  $l$  be the index of

the iteration, and let  $\bar{\phi}$  be an intermediate result. Then the following computational scheme is proposed:

$$\begin{aligned} L\bar{\phi} &= \mathbf{b}(x, y, \phi^l) \\ L\phi^{l+1} &= \mathbf{b}(x, y, \bar{\phi}). \end{aligned} \quad (18)$$

Formally, Eq. (18) is equivalent to

$$L\phi^{l+1} = \tilde{\mathbf{b}}(x, y, \phi^l). \quad (19)$$

To start the iteration, we use the value of the field variables at the  $r^{th}$  level in  $x$  (i.e.,  $\phi^0 = \phi_r$ ). The condition for convergence of Eq. (18) is found by appealing to the fixed-point theorem.

For the purpose of determining the convergence criterion, define  $\mathcal{C}$ , a region in  $\mathcal{C}^4$ , the generalization of the four-dimensional real space to complex variables. Let  $\Phi$  and  $\mathbf{h} \in \mathcal{C}$  be two vectors in that space. Hence, the derivative of  $\mathbf{A}$  with respect to  $\Phi$  is defined as

$$\mathbf{A}_{\Phi} \equiv \mathbf{J}(\Phi) = \frac{\partial A_i}{\partial \Phi_j}. \quad (20)$$

If the second derivative is continuous for all  $\Phi \in \mathcal{C}$ , then it satisfies

$$\| \mathbf{A}_{\Phi\Phi}(\Phi, \mathbf{h}, \mathbf{h}) \| \leq R \| \mathbf{h} \|^2 \quad (21)$$

for all  $\Phi$ .

Furthermore, let  $\|\cdot\|_p$ , with  $p = 1, 2, \infty$ , represent the induced norms

$$\begin{aligned} l_1 &= \max_{1 \leq j \leq n} \{ \sum_{i=1}^n |A_{ij}| \} & l_2 &= \{ \sum_{i=1}^n \sum_{j=1}^n A_{ij} A_{ij}^* \}^{1/2} \\ l_\infty &= \max_{1 \leq i \leq n} \{ \sum_{j=1}^n |A_{ij}| \}. \end{aligned} \quad (22)$$

Finally, define a supersystem on  $\mathcal{R}_\Delta^2$  as

$$[\partial_x - \mathbf{K} \partial_{yy} + \mathbf{K}_f] \Phi = \mathbf{B}(x, y, \Phi) \quad (23)$$

plus boundary conditions,

$$\begin{aligned} \Phi_y &= 0 & \text{on } y=0, y=N \\ \Phi &= \Phi_0 & \text{on } x=0 \end{aligned} \quad (24)$$

composed of (13) and its complex conjugate, with

$$\begin{aligned} \mathbf{K} &= [\mathbf{k}, \mathbf{k}^*]^T \in \mathcal{C} \\ \mathbf{K}_f(x, y) &= [\mathbf{k}_f, \mathbf{k}_f^*]^T \in \mathcal{C} \\ \Phi &= [a_1(x, y), a_2(x, y), a_1^*(x, y), a_2^*(x, y)]^T \in \mathcal{C}. \end{aligned} \quad (25)$$

Let  $\mathbf{L}$  be the resulting discrete operator of the supersystem, composed of  $L$  and its complex conjugate. Choosing a nonsingular representation for  $L$  (hence  $\mathbf{L}$  will be nonsingular as well), and multiplying both sides of (23) by  $\mathbf{L}^{-1}$ , we have

$$\Phi^{l+1} = \mathbf{A}(x, y, \Phi^l). \quad (26)$$

Define the iteration discrepancy as

$$\|\delta\Phi^{l+1}\|_p \equiv \|\Phi^{l+1} - \Phi^l\|_p. \quad (27)$$

Appealing to the fixed-point theorem, we can surmise that

$$\begin{aligned} \|\delta\Phi^{l+1}\|_p &= \|\mathbf{A}(\Phi^l) - \mathbf{A}(\Phi^{l-1})\|_p \\ &\approx \|\mathbf{J}(\Phi^{l-1})\delta\Phi^l\|_p \leq \|\mathbf{J}(\Phi^{l-1})\|_p \|\delta\Phi^l\|_p \\ &\leq \|\mathbf{J}(\Phi^{l-1})\|_p \|\mathbf{J}(\Phi^{l-2})\|_p \|\delta\Phi^l\|_p \leq \dots \\ &\leq \prod_{l=0}^{k-1} \|\mathbf{J}(\Phi^l)\|_p \|\delta\Phi^0\|_p, \end{aligned} \quad (28)$$

provided

$$0 < \|\mathbf{J}(\Phi^l)\|_p < 1. \quad (29)$$

Equation (29) is in fact the convergence criterion for the iteration process.

To establish an estimate of the rate of convergence, let  $r > 0$  be given such that the set of vectors  $\mathcal{S} = \{\Phi : \|\Phi - \mathbf{s}\|_p < r\}$  contains a fixed point  $\mathbf{s}$  of  $\mathbf{A}(\mathbf{s})$ , that is,

$$\mathbf{s} = \lim_{l \rightarrow \infty} \Phi^l = \lim_{l \rightarrow \infty} \mathbf{A}(\Phi^l) = \mathbf{A}(\mathbf{s}). \quad (30)$$

Further, let  $\mathcal{S} \subseteq \mathcal{C}$ ,  $\mathbf{J}(\mathbf{s})$  continuous on  $\mathcal{S}$  and  $\|\mathbf{J}(\mathbf{s})\|_p < 1$ . Then there exists an  $\varepsilon > 0$  such that the fixed point iteration is convergent whenever  $\|\Phi^0 - \mathbf{s}\|_p < \varepsilon$ . Define  $\|\mathbf{e}^{l+1}\|_p$ , the measure of difference between the  $(l+1)^{th}$  iterate and the

root. Hence

$$\| \mathbf{e}^{l+1} \|_p = \| \Phi^{l+1} - \mathbf{s} \| \approx \| \mathbf{J}(\mathbf{s})\mathbf{e}^l + \mathbf{A}''(\zeta; \mathbf{e}^l, \mathbf{e}^l) \|_p \leq \| \mathbf{J}(\mathbf{s})\mathbf{e}^l \| + R \| \mathbf{e}^l \|^2 . \quad (31)$$

Quadratic convergence is possible if  $\mathbf{J}(\mathbf{s}) = 0$ ,

$$\lim_{l \rightarrow \infty} \frac{\| \mathbf{e}^{l+1} \|_p}{\| \mathbf{e}^l \|_p} \leq R. \quad (32)$$

For the problem in question, however, the best rate of convergence will be linear since  $\mathbf{J}(\mathbf{s}) \neq 0$ :

$$\lim_{l \rightarrow \infty} \frac{\| \mathbf{e}^{l+1} \|_p}{\| \mathbf{e}^l \|_p} \leq \| \mathbf{J}(\mathbf{s}) \|_p . \quad (33)$$

A measure of resources required in the computation is the size of the resulting matrix problem. The slightly better flexibility in the choice of discretization for the linear operator  $L$  is the key advantage of this method over others. The most economical discretizations are those that lead to a tridiagonal or pentadiagonal matrix. The choice

$$L = \left( \frac{3}{2\Delta x} \Delta_x - \frac{1}{2\Delta x} \nabla_x \right) \phi_r^s - \left( \frac{\mathbf{k}}{\Delta y^2} \delta_y^2 + \mathbf{k}_{\mathbf{f}_{r+1}} \right) \phi_{r+1}^s \quad (34)$$

leads to an  $n \times n$  tri-diagonal matrix. This choice of discretization is commonly known as the Douglas scheme.  $L$  has eigenvalues

$$\lambda_s = -(3 + 2\rho + 2\Delta x \mathbf{k}_{\mathbf{f}}) + 2\rho \cos\left[\frac{s\pi}{n+1}\right] \quad s = 1, \dots, n , \quad (35)$$

and the eigenfunctions are given by

$$\left\{ \sin \frac{s\pi}{n+1}, \sin \frac{2s\pi}{n+1}, \dots, \sin \frac{s\pi}{n+1} \right\}^T \quad s = 1, \dots, n. \quad (36)$$

Furthermore, the operator  $L$  is diagonally dominant, since

$$\sum_{j \neq i}^{n+1} |L_{ij}| \leq |L_{ii}| \quad i = 1, \dots, n, \quad (37)$$

the  $L_{ij}$ 's being the entries of the matrix  $L$ , and nonsingular since

$$\begin{aligned} |L_{ii}| &> |L_{ii+1}| > 0 & i = 1, \dots, n-1 \\ |L_{ii}| &\geq |L_{ii+1}| + |L_{ii-1}| \quad L_{ii+1}L_{ii-1} \neq 0 & i = 2, \dots, n-1 \\ |L_{ii}| &> |L_{ii-1}| & i = 2, \dots, n. \end{aligned} \quad (38)$$

If  $\phi = \xi^r e^{i\theta}$ , where  $\theta = \alpha \Delta y s$ , and  $\rho = 2 \frac{\Delta x}{\Delta y^2} \mathbf{k}$ , upon substituting these quantities in  $L$  the magnification factor is

$$\xi = \frac{1}{2\rho(1 - \cos \theta) + 2\Delta x \mathbf{k}_f + 3} \left\{ 2 \pm \sqrt{1 - 2\rho(1 - \cos \theta) - 2\Delta x \mathbf{k}_f} \right\}, \quad (39)$$

from which it is clear that  $|\xi| \leq 1$ . Thus the linear operator is unconditionally stable.

An estimate of the accuracy of the discretization of the linear operator, as well as a check on its consistency with the continuous operator on the grid, is given by

$$(L - \mathcal{L})\phi = \frac{\Delta x^2}{2} \phi_{xx} + \mathbf{k} \frac{\Delta y^2}{12} \phi_{yyyy} + \dots, \quad (40)$$

where  $\mathcal{L}$  is the continuous linear operator. Equation (40) implies that the scheme is  $O(\Delta x^2 + \Delta y^2)$  accurate.

This order of accuracy is an upper bound on the accuracy of the overall scheme; hence, attempting to reduce the error  $\| \mathbf{e}^l \|$  much below this is pointless. Since the root is not known a priori, the iteration procedure is carried out until the discrepancy is safely below the above-quoted error, but not much beyond that. This, in effect, is the criterion used in the code for stopping the root-finding iteration procedure.

Consistency of the discretization is readily established by comparing the continuous problem with its discretization in the limit as the grid size gets smaller. It can be shown that the discretization approaches the continuous operator on the grid uniformly.

Since the Douglas scheme is inapplicable at  $r = 0$ , a standard backwards Euler scheme,

$$\frac{1}{\Delta x} \Delta_x \phi_r - \frac{1}{\Delta y^2} \mathbf{k} \delta^2 \phi_{r+1} + A_x(\mathbf{k}_f \phi)_{r+1}, \quad (41)$$

is used to discretize  $L$  for the first step in  $x$ , which can be shown to be unconditionally stable as well.

Having made a choice on the particular form of the operator  $L$ , the condition that  $\| \mathbf{J}(\mathbf{x}) \|_p < 1$  for the surface system must be determined explicitly, so that convergence is established for the sand ridge problem. To estimate the size of  $\mathbf{J}(\mathbf{x})$  we use the supersystem, Eq. (26), to find that

$$\delta \Phi^{l+1} \approx \mathbf{J}(\Phi^l) \delta \Phi^l \quad (42)$$

$$\delta\Phi^l \approx \mathbf{J}(\Phi^{l-1})\delta\Phi^{l-1} \quad (43)$$

$$\text{etc.} \quad (44)$$

with

$$\mathbf{J} = \mathbf{L}^{-1}\mathbf{B}'(\Phi), \quad (45)$$

where

$$\mathbf{B}'(\Phi^l) = \begin{pmatrix} 0 & -iK_5e^{-i\delta_{x_{r+1}}}a_1^{*l} & -iK_5e^{-i\delta_{x_{r+1}}}a_2^l & 0 \\ -i2K_6e^{-i\delta_{x_{r+1}}}a_1^l & 0 & 0 & 0 \\ iK_5e^{+i\delta_{x_{r+1}}}a_2^{l*} & 0 & 0 & iK_5e^{+i\delta_{x_{r+1}}}a_1^l \\ 0 & 0 & i2K_6e^{+i\delta_{x_{r+1}}}a_1^{l*} & 0 \end{pmatrix}, \quad (46)$$

for the  $l^{th}$  iterate. In Eq. (46), it is understood that the  $a$ 's are defined only on the grid.

From Eq. (45),  $\|\mathbf{J}\|_p < 1$  if the size of  $\mathbf{L}$  is greater than the size of  $\mathbf{B}'$ . In the  $l^2$  norm, the convergence condition is

$$\|\mathbf{J}\|_2 = \|\mathbf{L}^{-1}\mathbf{B}'\|_2 \leq \|\mathbf{L}^{-1}\|_2 \|\mathbf{B}'\|_2 \leq 1. \quad (47)$$

Since  $\mathbf{L}\mathbf{L}^\dagger = \mathbf{L}^\dagger\mathbf{L}$ , where  $\mathbf{L}^\dagger$  is the Hermitian matrix of  $\mathbf{L}$ , then the spectral radius gives a measure of the two norm. Using this information, we have that

$$\|\mathbf{L}^{-1}\|_2 \leq \min_{s=1,n} |\lambda_s|, \quad (48)$$

or, using Eq. (35),

$$+ \sqrt{(K_5^2 + 4K_6^2)|a_1|^2 + K_5^2|a_2|^2} / (3 + 2\Delta x \mathbf{k}_f) < 1, \quad (49)$$

where the  $l_\infty$  in  $y$  is used to estimate the size of the vectors, that is,  $a_i = \max_{1 \leq s \leq n} a_i^s$ ,  $i=1,2$ . Hence, Eq. (49) gives strict constraints on  $\rho$ ,  $\Delta x$ , and  $a_i$ , to be satisfied in order to guarantee convergence in the solution. Another constraint imposed in the numerical implementation was to restrict  $\Delta x$  to be less than  $2\pi/\delta$ , so as to minimize the phase error. In Figures 14 and 15 the parametric plots of  $K_5$  and  $K_6$  are shown; they complete the picture of the relevant size of all the quantities involved in Eq. (49).

## 4 Performance Evaluation of the Numerical Schemes

### 4.1 Evaluation of the Mass Transport Equation Scheme

We ran a few test cases in order to confirm qualitatively the stability, consistency and accuracy of the Lax-Wendroff scheme, checking for agreement with the well-established theoretical results. Of more concern to us was the issue of damping and of phase drift. To quantify the scheme's dissipation and drift, we used a model problem for which an exact solution is known.

The model problem was

$$h_T + kh h_x = 0, \quad x \in \mathcal{R}^1, T > 0, \quad (50)$$

with initial condition

$$h(x, 0) = \begin{cases} 1 & x < 0 \\ 1 + \varepsilon x & 0 \leq x \leq l \\ 1 + \varepsilon l & x > l \end{cases} \quad (51)$$

in which  $0 < k < 1$ , and  $\varepsilon < 1$ . The exact solution of Equations (50) and (51) is

$$h(x, T) = \begin{cases} 1 & x < kT \\ 1 + \varepsilon \frac{x - kT}{1 + \varepsilon kT} & kT \leq x \leq l + k(1 + \varepsilon l)T \\ 1 + \varepsilon l & \text{otherwise.} \end{cases} \quad (52)$$

Different values of  $k$  were tried—it scales the time step—but we report our results for  $k = 0.1$ . For such a case, convergence is possible if  $h\Delta T/\Delta x < 10$  in the time interval 0 to  $T$ . Since Eq. (50) conserves a quantity proportional to  $h^2$ , we compared the computed value  $h_\Delta$  with the theoretical value  $h$  as a function of  $a \equiv k\Delta T/\Delta x$  and as a function of time  $T$ , to get an idea of the scheme's dissipation. Specifically, we monitored the constant of motion

$$c(T, a) = \sum_{r=0}^{M/\Delta x} h_\Delta^2(T, r\Delta x)r\Delta x + \frac{2}{3}kT[h_\Delta^3(T, M) - h_\Delta^3(T, 0)], \quad (53)$$

where  $M$  is a very large value in  $x_r$ . For an estimation of the phase drift, we computed

$$e^2(a, T) = \sum_r |h_\Delta(T, x_r) - h(T, x_r)|^2 / \sum_r |h(T, x_r)|^2. \quad (54)$$

Figures 1 and 2 show parametric plots of  $c$  and  $e^2$ , respectively.

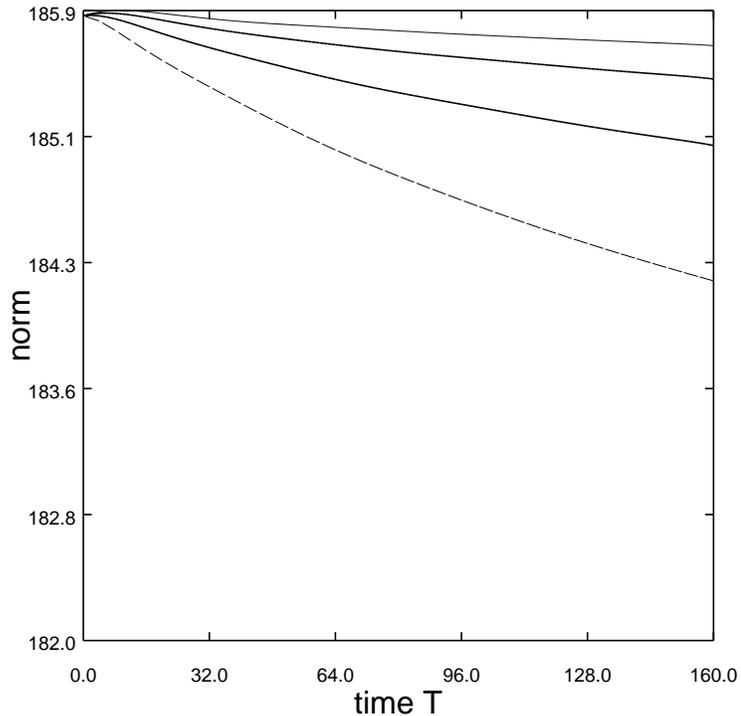


Figure 1: Dissipation as a function of  $a$  and  $T$  with  $k = 0.1$  for the Lax-Wendroff scheme. From top to bottom,  $a = 0.4, 0.2, 0.1, 0.05$ , respectively.

## 4.2 Performance of the Runge-Kutta Scheme

The accuracy and dissipation of the explicit fourth-order Runge-Kutta was investigated using a flat bottom and  $\mathcal{A}_j$  constants. The domain was 128 units in extent, or roughly 10 interaction lengths. In this special case it may be shown that the energy is given by the sum of the mode amplitudes squared. This quantity was conserved by all trials to within 2% for all reasonable grid sizes.

An exact solution to Eq. (11) is known when  $f(x) = 0$ . To estimate the

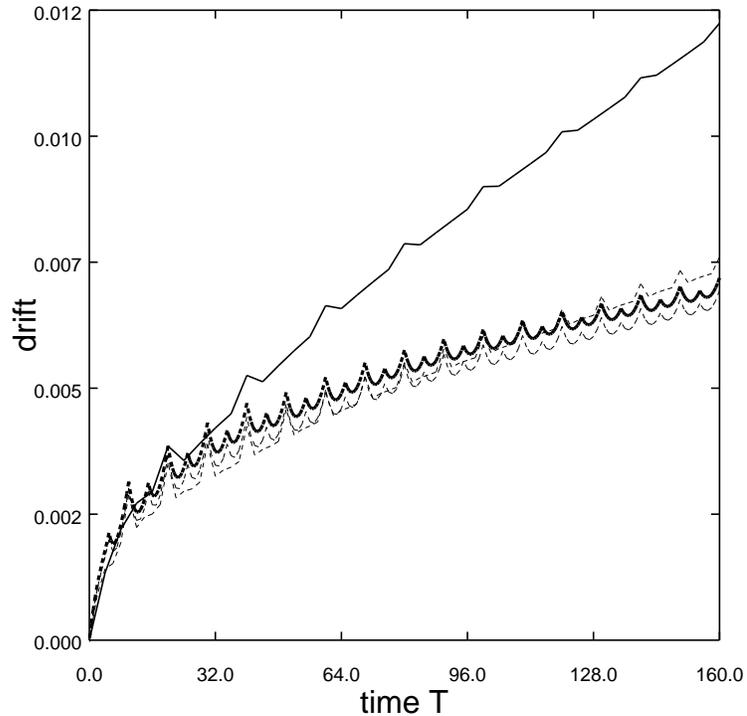


Figure 2: Phase drift for the Lax-Wendroff scheme as a function of  $a$  and  $T$  with  $k = 0.1$ . From top to bottom,  $a = 0.4, 0.2, 0.1, 0.05$ , respectively.

accuracy and error of the scheme, we compared the outcome of the numerical solution with the exact solution of this special case [5], which, in terms of Jacobi elliptic functions “sn,” are

$$\begin{aligned}
 v^2(\tilde{x}) &= v_a^2 + (v_b^2 - v_a^2) \operatorname{sn}^2[(v_c^2 - v_a^2)^{1/2}(\tilde{x} + \tilde{x}_0); \gamma] \\
 w^2(\tilde{x}) &= 1 - v^2(\tilde{x}).
 \end{aligned} \tag{55}$$

The quantity  $\tilde{x}_0$  is determined by the boundary condition  $y^2(0)$ , and the value of  $\gamma$  is related to the three roots  $v_a, v_b$ , and  $v_c$ . The following measures were used to

estimate the accuracy and error:

$$\begin{aligned}
l_\infty &= \frac{\max\{\sum_r |\chi(r\Delta x) - \chi'(x_r)|\}}{\max\{\sum_r |\chi'(r\Delta x)|\}} \\
l_1 &= \frac{\sum_r |\chi(r\Delta x) - \chi'(x_r)|}{\sum_r |\chi'(r\Delta x)|} \\
l_2 &= \frac{[\sum_r |\chi(r\Delta x) - \chi'(x_r)|]^2]^{1/2}}{[\sum_r |\chi'(r\Delta x)|^2]^{1/2}}, \tag{56}
\end{aligned}$$

where  $\chi$  is the calculated value of  $a_i$ , and  $\chi'$  the exact value at the grid location. The exact solution  $\chi'$ , was computed using the algorithm given in [6], p. 189. The error as a function of grid size is plotted in Figure 3, from which one can conclude that the scheme is in fact fourth-order accurate and consistent. For the accuracy and dissipation trials  $\mathcal{A}_1 = 0.5$ ,  $\mathcal{A}_2 = 0$ , in Eq. (11), a flat bottom and parameters  $\alpha = 0.3$ ,  $\beta = 0.1$ ,  $\omega_1 = 0.5$ , were used.

### 4.3 Fixed-Point Method Performance and Evaluation

Since an exact solution to the three-dimensional internal wave system is as yet unknown, we sought to discern the accuracy of the fixed-point method (FPM) using local analysis. Let  $\Delta$  be the size in  $x$  or  $y$  of each grid element. A comparison of the computed solution at a particular point, using  $\Delta$ , with a solution with grid size  $\Delta/2$  yields

$$|\chi_\Delta - \chi_{\Delta/2}| \equiv k_1 = CO'[(\Delta/2)^p]. \tag{57}$$

Halving the grid size again

$$|\chi_{\Delta/2} - \chi_{\Delta/4}| \equiv k_2 = CO[(\Delta/4)^p]. \tag{58}$$

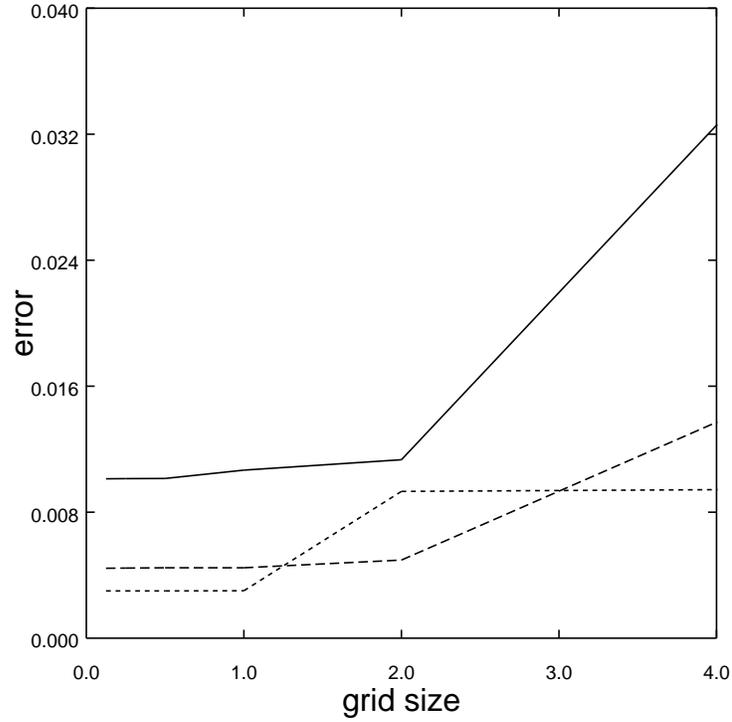


Figure 3: Error as a function of grid size for the Runge-Kutta method.  $l_1$ :  $\cdots\cdots\cdots$   
 $l_2$ :  $-\ -\ -\ -\ -\ -$ ,  $l_\infty$ :  $\text{—————}$ .

Thus, using Eq. (57) and Eq. (58) one can solve for  $p$  to get an estimate of the order of accuracy of the scheme:

$$p = \frac{\log k_1 - \log k_2}{\log 2}. \quad (59)$$

Using the same parameters and boundary conditions as those used in connection with the Runge-Kutta scheme evaluation trials, and a domain with length of 128 and span of 32, we found that FPM yields an average value of  $p = 1.8$ , with a standard deviation of 0.5. Values of both field quantities were used to estimate  $p$ ,

and they were taken from various points in the domain.

A systematic study of the convergence of the method was not carried out. However, the computed values tended to converge as the grid size was refined. Since comparisons of the computed solutions with an exact expression for the three-dimensional case were not possible, a comparison of the cross-sectional values of an effectively two-dimensional solution computed using FPM along the whole length in  $x$  and midway in the spanwise direction  $y$ , with a solution computed using the Runge-Kutta method with a very fine grid spacing was made to ascertain the qualitative correctness of the FPM scheme. A measure of the error is given by the norms

$$\begin{aligned}
 l_{\infty}(\Delta x, \Delta y) &= \frac{\max\{\sum_r |\chi(r\Delta x, mid) - \chi'(x_r)|\}}{\max\{\sum_r |\chi'(r\Delta x)|\}} \\
 l_1(\Delta x, \Delta y) &= \frac{\sum_r |\chi(r\Delta x, mid) - \chi'(x_r)|}{\sum_r |\chi'(r\Delta x)|} \\
 l_2(\Delta x, \Delta y) &= \frac{[\sum_r |\chi(r\Delta x, mid) - \chi'(x_r)|]^2]^{1/2}}{[\sum_r |\chi'(r\Delta x)|^2]^{1/2}}, \tag{60}
 \end{aligned}$$

where  $\chi$  represents the solution obtained using FPM and  $\chi'$  the solution computed with the Runge-Kutta scheme.

The result for the case  $\Delta x = \Delta y$  is shown in Figure 4. The same outcome is obtained when  $\Delta y = 0.25$  and  $\Delta x$  is varied. On the other hand, when  $\Delta x = 0.25$  is fixed and  $\Delta y$  is varied, very little change in the norms is observed. In this last case, the norms had an approximate value of  $4.4 \times 10^{-3}$  for all grid sizes in the  $y$  direction that were used. Note that there is no  $y$  dependence in the solution for this particular trial.

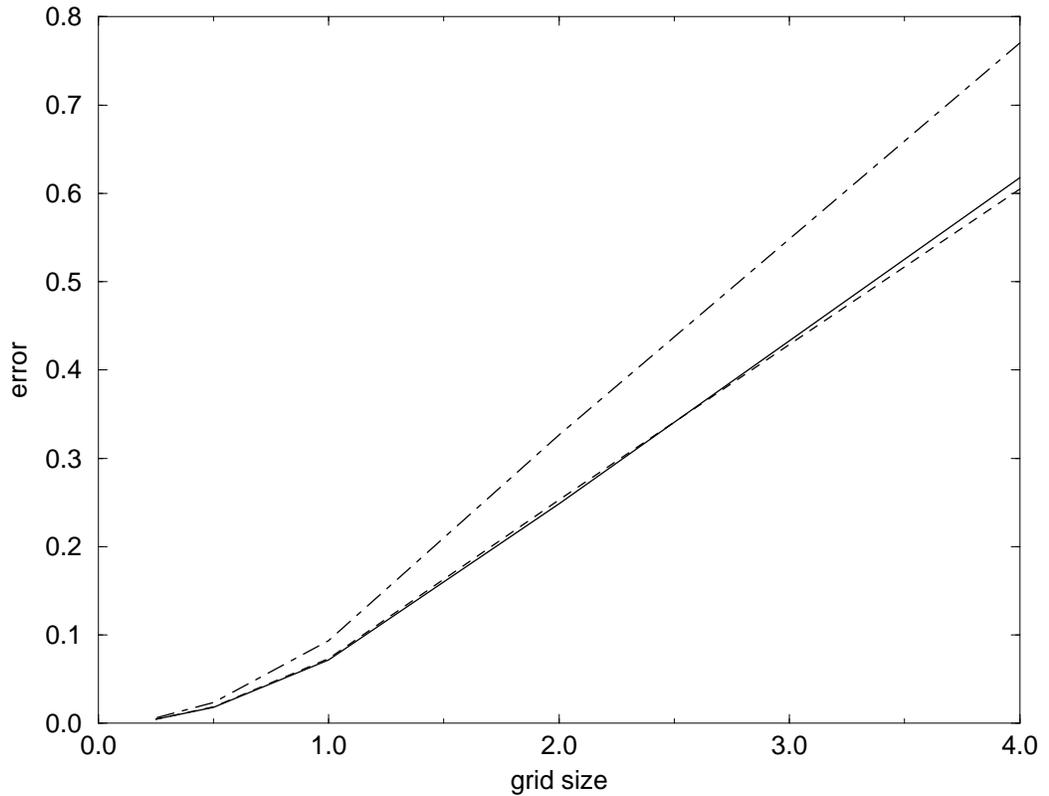


Figure 4: Error as a function of grid size, with  $\Delta x = \Delta y$ .  $l_\infty$ : - · - · - · -,  $l_1$ : ·······,  $l_2$ : —————.

The rate at which the iteration procedure converges in FPM as a function of the grid size was also investigated. With  $\alpha = 0.3$ ,  $\beta = 0.08$ ,  $\omega_1 = 0.5$ , and boundary conditions  $\mathcal{A}_1 = 0.5$  and  $\mathcal{A}_2 = 0.1$ , and a flat bed, the iteration discrepancy

$$\log_{10}[\max\{\sum_{s=0}^n |\phi^{l+1}(x, s\Delta y) - \phi^l(x, s\Delta y)|\}] \quad (61)$$

was monitored at a particular value of  $x$  in a fairly large domain. As expected, it was found that the number of iterations required to meet a certain iteration tolerance decreased as the grid was refined. Figure 5 shows how the iteration

discrepancy drops after each iteration  $l$  for a number of different grid sizes. It is evident from the graph that a finite and small number of iterations are required to reach adequate error tolerances using reasonably-sized grids.

The iteration convergence of the solution at the first step in  $x$  was examined as well. Recall that for the first step a backwards Euler scheme was used to discretize the linear operator instead of the Douglas scheme. The finding is that the number of iterations was roughly double the number required elsewhere in the domain, where the Douglas scheme is used.

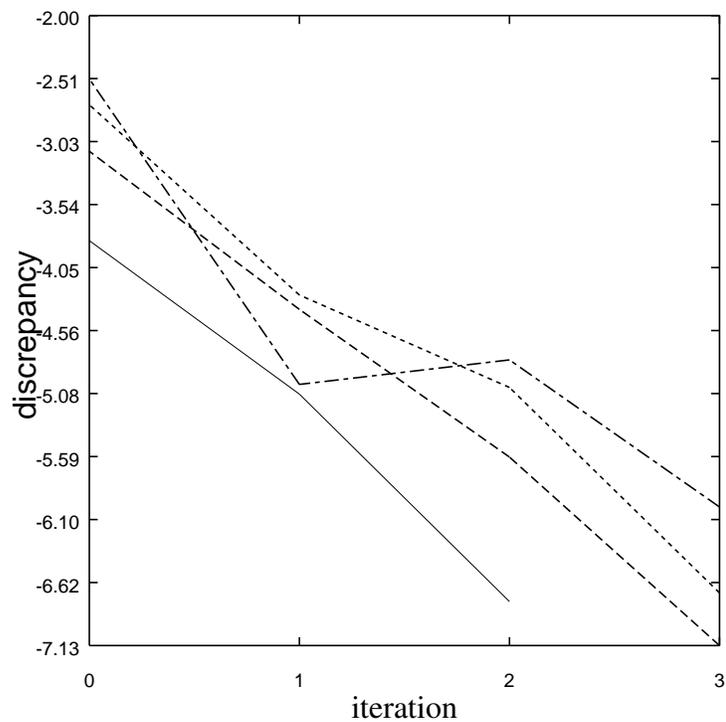


Figure 5: Iteration discrepancy as a function of grid spacing. The number of iterations drops as  $\Delta = 4, 2, 1, 0.5$  respectively.

#### 4.4 Storage and Speed of the FPM

An estimate for the operation count for the FPM is as follows. Eq. (18) leads to the problem

$$L\phi = \mathbf{b} \quad (62)$$

for the unknown  $\phi$ , where  $L$  is a  $2n \times 2n$  tridiagonal matrix,  $m$  times to cover all values of  $x$  in the domain. The efficient way to solve Eq. (62) is to decompose the problem in two steps: let  $L = WU$ , where  $W$  is a lower triangular matrix and  $U$  and upper triangular matrix. Then

$$W\mathbf{g} = \mathbf{b} \quad (63)$$

is solved for  $\mathbf{g}$ , followed by

$$U\phi = \mathbf{g}, \quad (64)$$

to finally obtain  $\phi$ . The total operation count for the solution of Equations (63) and (64) is  $(5n - 4)$  multiplies and  $(3n - 3)$  adds. All told,  $O(16n)$  operations. In turn, this process is performed  $l$  times to compute the  $(l + 1)^{th}$  iterate, and finally  $m$  times to cover all values of  $x$ . The total is  $m \times l \times O(n)$ .

The storage requirements of the FPM may be estimated as follows: the old and the new vector at each  $x$ , and another vector for the iteration process, need to be stored. Hence  $6n$  values are stored. In addition, all the entries of a tridiagonal matrix of size  $2n \times 2n$ , or roughly  $6n$  values need to be stored. The total is thus  $12n$ , or  $O(n)$  values. In fact, we could be even more economical and use multipliers in the entries of  $L$ , so that only one half of the tri-diagonal matrix entries need to

be stored.

Note that its economy of resources hinges upon the simplicity of the matrix that the discretization generated. If higher-order accuracy is required, the matrix will be more complicated than the simple tridiagonal matrix that was used in this study, requiring greater computational resources. A somewhat unavoidable problem with the FPM is that the discretization has significant dissipation. However, the dissipation can be made tolerable at the expense of greater computational resources, that is, by refining the grid.

To illustrate the degree of dissipation in the surface system FPM implementation, we used the same parameters and domain that was used in connection with the iteration issue, and we fixed the iteration discrepancy tolerance at  $10^{-6}$ . Two types of trials were carried out, both using a flat bottom. In the three-dimensional trial we assumed the boundary conditions were  $\mathcal{A}_1 = 0.5 + 0.01y$  and  $\mathcal{A}_2 = 0.1 + 0.01y$  and monitored the conserved Hamiltonian [5] along the length in the  $x$  direction, midway in the spanwise direction. In the two-dimensional trial, we set  $\mathcal{A}_1 = 0.5$  and  $\mathcal{A}_2 = 0.1$  and monitored the same quantity along the ray. The outcome of both trials was qualitatively similar: the computed conserved quantity oscillated with a period equal to the interaction length. The difference between the peak value and the minimum value increased as the grid size was made larger. In addition, dissipation (i.e., the drop of the peak value as a function of position  $x$ ) increased as the grid size was made larger, and as a result, the local interaction length grew since the amplitude of the modes were attenuated. The attenuation we know is inherent in the discretization of the linear operator. The dissipation and oscillation of the conserved quantities can be made negligible by refining the grid. We also

<i>Grid Size <math>\Delta</math></i>	<i>Fluctuation</i>
4.00	0.1002
2.00	0.0627
1.00	0.0168
0.50	0.0050
0.25	0.0014

Table 1: Energy fluctuation vs. grid size. Equilateral grid case.

<i>Grid Size <math>\Delta y</math></i>	<i>Fluctuation</i>
4.00	0.0018
2.00	0.0013
1.00	0.0013
0.50	0.0012
0.25	0.0014

Table 2: Energy fluctuation vs.  $\Delta y$ .  $\Delta x = 0.25$  fixed.

found that the effect is much more pronounced when  $\mathcal{A}_2 = 0$  exactly, which yields solutions with very sharp minimas in the field variables. Table 1 shows the difference between successive maxima and minima for the second trial as a function of grid size, with  $\Delta x = \Delta y$ . We also report the outcome of fixing  $\Delta x = 0.25$  and varying  $\Delta y$ , in Table 2, and the opposite settings are illustrated in Table 3. The two-dimensional trials for  $\Delta x = 0.25$  and  $\Delta y = 4$  showed significant discrepancies when compared with the Runge-Kutta calculation, and the energy for this case oscillated in a somewhat regular pattern.

<i>Grid Size <math>\Delta x</math></i>	<i>Fluctuation</i>
4.00	0.1415
2.00	0.0628
1.00	0.0198
0.50	0.0049
0.25	0.0014

Table 3: Energy fluctuation vs.  $\Delta x$ .  $\Delta y = 0.25$  fixed.

To conclude this section, we report the wall-clock times for three runs of the internal wave equations, as discretized using FPM. The code was written in Fortran 77—because of issues related to code portability—in a straightforward manner, except that recursion was used in the iteration procedure. For the size of these runs, the use of recursion was probably marginally slower than having opted for repeated subroutine calls. No machine optimization or floating-point accelerators were used.<sup>1</sup> The time trials were carried out with an initial bottom configuration  $f = 0.01x$ . All other parameters and physical quantities were the same as those used previously. The domain was a square with 50 units to its side. Two times are reported, the first one, in Table 4, corresponds to the total time required to find the field variables everywhere in the domain, and a second one, given in Table 5, is the time required to compute all values in the  $y$  direction, for a particular  $x$ .

---

<sup>1</sup>The Titan’s vectorizability was not exploited either. Otherwise, its reported performance would not compare so unfavorably.

Machine	$\Delta = 1, (50 \times 50)$	$\Delta = 0.5, (100 \times 100)$	$\Delta = 0.25, (200 \times 200)$
Sun Sparc SLC	7.43	25.42	78.8
Sun Sparc 2	2.29	7.81	23.13
Ardent Titan 2X P1	3.9	13.9	44.81

Table 4: Wall-clock times in seconds vs. grid size (number of grid points per domain) for the computation of the surface system over the whole domain using the fixed-point method

Machine	$\Delta = 1, (50 \times 50)$	$\Delta = 0.5, (100 \times 100)$	$\Delta = 0.25, (200 \times 200)$
Sun Sparc SLC	0.16	0.25	0.50
Sun Sparc 2	0.06	0.08	0.15
Ardent Titan 2X P1	0.08	0.13	0.29

Table 5: Wall-clock times in seconds for the computation of the surface system for all values of  $y$  at a particular  $x$  using the fixed-point method

## 5 Summary

The model for the formation and evolution of three-dimensional sand ridges on the continental shelf described in [1] and [2] has been shown to be adequately discretized using finite difference techniques and fixed point methods. The mass transport equation is implemented by using a standard Lax-Wendroff scheme, while the surface system was discretized using a Douglas scheme for the linear part and iterative correction for the nonlinear terms. We call such a scheme the fixed-point method (FPM).

The schemes' performance was evaluated in detail. It was found that both schemes are second-order accurate in time and space. They were also found to be efficient in both storage and speed and quite straightforward in their computer implementation. The schemes were found to converge as the mesh size was diminished.

The Lax-Wendroff scheme was found to have significant phase drift, especially when the mesh size is increased. The FPM was shown to have significant diffusion for large grid spacings. This damping will introduce phase errors in the waves, especially if the domain is quite large.

Included in this study is a prescription to monitor the stability of the solutions. This condition was monitored in all trial runs a posteriori. The condition poses a severe restriction on the size of the computed solutions, but it has been found to be large enough to encompass most physically relevant situations.

In order to not introduce unwanted symmetries in the solution of the wave system, a "zero flux condition" was introduced to handle the boundary conditions on

the lateral sides of the domain. Briefly described, the condition amounts to placing Neumann boundary conditions on the lateral sides of the domain, sufficiently far away from the region of interest, connecting this domain to lateral swaths of computational space in which the three-dimensionality of the solutions is gradually collapsed into two dimensions. While the technique was not a resounding success, especially for domains that were very long in the  $x$  direction in which ample space was available for the solution in the region of interest to be affected by the hard lateral barriers, it was preferred over other alternatives that would complicate the problem or pose severe symmetry conditions on the solutions.

In summary, the solutions to the model may be confidently found using the techniques described in this paper. In order to reduce the phase error and dissipation in the computed solutions, the mesh size must be small. The size of the solutions to the wave system must be monitored to insure stability. The regime of stability of the mass transport equation discretization was found to be well estimated by the well-known Courant-Friedrich-Lewy condition.

## Acknowledgments

We thank the Applied Research Laboratory at The Pennsylvania State University for making this project possible. This work was also supported by the Office of Scientific Computing, U.S. Department of Energy, under Contract W-31-109-Eng-38.

## A Appendix

The following are constants associated with Eq. (1):

$$\begin{aligned}
 K_1 &= F_1 \\
 K_2 &= F_2 \\
 K_3 &= D_1 E_1 \\
 K_4 &= D_2 E_2 \\
 K_5 &= D_1 S_1 \\
 K_6 &= D_2 S_2
 \end{aligned}
 \tag{65}$$

with

$$\begin{aligned}
 D_j &= [2(1 - \beta^2 \frac{\omega_j^2}{3})]^{-1} \\
 E_j &= k_j (1 - \frac{2}{3} \beta^2 \omega_j^2) \\
 F_j &= 1/2 k_j \\
 S_1 &= \frac{k_2 - k_1}{\omega_1} \{k_2 - k_1 + \omega_1 (\frac{\omega_1}{k_1} + \frac{\omega_2}{k_2})\} \\
 S_2 &= 2(k_1^2 / + 2\omega_1^2) / \omega_2.
 \end{aligned}
 \tag{66}$$

## List of Figures

- 1 Dissipation as a function of  $a$  and  $T$  with  $k = 0.1$  for the Lax-Wendroff scheme. From top to bottom,  $a = 0.4, 0.2, 0.1, 0.05$ , respectively. . . . . 21
- 2 Phase drift for the Lax-Wendroff scheme as a function of  $a$  and  $T$  with  $k = 0.1$ . From top to bottom,  $a = 0.4, 0.2, 0.1, 0.05$ , respectively. 22
- 3 Error as a function of grid size for the Runge-Kutta method.  $l_1$ :  
 .....  $l_2$ : - - - - - ,  $l_\infty$ : —————. . . . . 24
- 4 Error as a function of grid size, with  $\Delta x = \Delta y$ .  $l_\infty$ : - · - · - · - · - ,  $l_1$ :  
 .....,  $l_2$ : —————. . . . . 26
- 5 Iteration discrepancy as a function of grid spacing. The number of iterations drops as  $\Delta = 4, 2, 1, 0.5$  respectively. . . . . 27

## List of Tables

1	Energy fluctuation vs. grid size. Equilateral grid case. . . . .	30
2	Energy fluctuation vs. $\Delta y$ . $\Delta x = 0.25$ fixed. . . . .	30
3	Energy fluctuation vs. $\Delta x$ . $\Delta y = 0.25$ fixed. . . . .	31
4	Wall-clock times in seconds vs. grid size (number of grid points per domain) for the computation of the surface system over the whole domain using the fixed-point method . . . . .	32
5	Wall-clock times in seconds for the computation of the surface system for all values of $y$ at a particular $x$ using the fixed-point method	32

# References

- [1] J. M. Restrepo & J. L. Bona, “Model for the Formation of Longshore Sand Ridges on the Continental Shelf,” Mathematics and Computer Science Division, Argonne, Ill, December 1993.
- [2] J. M. Restrepo & J. L. Bona, “Model for the Formation of Longshore Sand Ridges on the Continental Shelf: The Interaction of Internal Waves and the Bottom Topography,” Mathematics and Computer Science Division, Argonne, Ill, December 1993.
- [3] J. D. Smith, *Numerical Solution of Partial Differential Equations, Finite Difference Methods*, Clarendon Press, Oxford, 1985.
- [4] B. Boczar-Karakiewicz, J. L. Bona & G. Cohen, *Interaction of Shallow-water waves and bottom topography*, PSU Applied Mathematics Series #AM3, Penn State University, 1986.
- [5] J. M. Restrepo & J. L. Bona, “Structure and Behavior of Triad Interactions for a Boussinesq System Arising in a Model for the Formation Sand Ridges,” Mathematics and Computer Science Division, Argonne, Ill, December 1993.
- [6] W. H. Press, B. P. Flannery, S. A. Teukolsky & W. T. Vetterling, *The Art of Scientific Computing, Numerical Recipes*, Cambridge University Press, Cambridge, 1989.